UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/031,260	05/17/2002	Manjunath M Suryanarayana	1419-136 US	6498

7590 10/07/2005	EXAMINER
Diana Dunn Mckay	RUTTEN, JAMES D
Mathews Collins Shepherd & Gould	

100 Thanet Circle Suite 306	ART UNIT	PAPER NUMBER
Princeton, NJ 08540	2192	

DATE MAILED: 10/07/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)
Office Action Summary	10/031,260	SURYANARAYANA, MANJUNATH M
	Examiner	Art Unit	
	J. Derek Rutten	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.
- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *17 May 2002*.

2a)☐ This action is **FINAL**.   2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-58* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-58* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *23 October 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All   b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____ .

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☒ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *6/6/02*.

4)☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____ .

DETAILED ACTION

1.   Claims 1-58 have been examined.

Claim Objections

2.   Claim 6 is objected to because of the following informalities: A typo appears as

"monotomically", which should likely be "monotonically". Appropriate correction is required.

3.   Claim 34 is objected to because of the following informalities: A typo appears as

"quiesent", which should likely be "quiescent". Appropriate correction is required.

Claim Rejections - 35 USC § 101

4.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or
> any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and
> requirements of this title.

5.   Claims 1-10, and 41-58 are rejected under 35 U.S.C. 101 because the claimed invention

is directed to non-statutory subject matter. The claim limitations recite functional descriptive

matter in terms of software technology including "objects", "object interface", and "object port".

These elements describe a software implementation, per se, and so is not tangible. The

"environment" of claims 1-10 would be statutory if the recited software elements are coupled

with hardware elements in order to provide a tangible implementation. Claims 41-58 are

directed to a "system" and suffer the same deficiencies as the "environment" of claims 1-10.

Claim Rejections - 35 USC § 112

6.   The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

7.   Claims 11-20 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

8.   Claim 11 recites "forwarding negotiations from said object ports to said management object ports" in line 14. The specification does not describe how a "negotiation" could be forwarded from one port to another. The word negotiation is a noun meaning "The act or process of negotiating" (The American Heritage® Dictionary of the English Language, Fourth Edition). In this context, a negotiation is not a discrete, singular event that could be "forwarded" from one port to another. The specification uses the word "negotiation" in this same context (e.g. page 23 lines 21-23: "...each step of the negotiation..."), and does not provide alternate definitions for the word that would support an act of "forwarding". However, page 23 lines 19-20 describes forwarding "negotiation scripts" through negotiation object ports. For the purpose of further examination, the word "negotiations" will be interpreted as "negotiation scripts".

9.   Claims 12-20 are rejected as being dependent upon a rejected base claim.

10.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

11.   Claims 1-20, 27-45 and 51-58 are rejected under 35 U.S.C. 112, second paragraph, as

being indefinite for failing to particularly point out and distinctly claim the subject matter which

applicant regards as the invention.

12.   Claim 1 recites the limitation "said ports having said compatible interfaces" in lines 9 and

10. There is insufficient antecedent basis for this limitation in the claim. Compatible interfaces

are recited in connection with objects, but are not recited in connection with object ports. This

limitation will be interpreted as "said ports having said compatible interfaces."

13.   Claim 4 recites "said objects" in line 3 and 6. It is not clear if this refers to earlier

recitations of "objects", or "management objects", or "human manager objects". Clarification is

required. Similar problems exist for claims 5, 7, 8, 10, 11, 12, 13, 27, 28, 29, 30, 32, 34, 35, 36,

37, 38, 39, 41, 42, 51, 53, 55, 56, and 57.

14.   Claim 6 recites the limitation "said application network" in line 3. There is insufficient

antecedent basis for this limitation in the claim. This limitation will be interpreted as "an

application network".

15.   Claims 1-10, and 41-58 are rejected under 35 U.S.C. 112, second paragraph, as being

incomplete for omitting essential elements, such omission amounting to a gap between the

elements. See MPEP § 2172.01. The omitted elements are: tangible computer hardware that

must form part of a computer-related system. These claims are all "system" claims, but do not

recite any such elements that form part of a tangibly embodied system.

16.   Claims 2-10, 12-20, 31-33, 42-45, 52-54, and 56-58 are rejected as being dependent upon

a rejected base claim.

Claim Rejections - 35 USC § 102

17.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the

basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on
sale in this country, more than one year prior to the date of application for patent in the United States.

18.   Claims 1-58 are rejected under 35 U.S.C. 102(b) as being clearly anticipated by

"OMSOFT: A Change Management Paradigm" Ph.D. dissertation by Suryanarayana (hereinafter

"OMSOFT").

In regard to claim 1:

1. A distributed object-oriented software development environment comprising:

a plurality of objects for performing object operations, each object including an

object interface;

at least one object port coupled to said each object interface of said objects;

interaction means for connecting said object port of one of said objects to said

object port of another one of said objects,

wherein one of said objects can communicate to another one of said objects if

said object interfaces are compatible and said interaction means provides sequential flow

of data and control from said object operations through a dynamically varying set of said

ports having said compatible interfaces.

OMSOFT discloses a distributed software development environment that consists

of objects connected via compatible interfaces connected using dynamic object ports for

communicating sequential flow of data and control. See Sections 3.2, 3.2.1, and 3.2.2 on

pages 40-42.

In regard to claim 2, OMSOFT further discloses:

2. The environment of claim 1 wherein said interaction means is represented by a

circular communication pathway and a first said object port is connected to said circular

communication pathway to receive communications from at least a second said object

port which is connected to said circular communication pathway. See page 45 and

Figure 3-2.

In regard to claim 3, OMSOFT further discloses:

3. The environment of claim 1 wherein said interface is described in modified CORBA

interface description language. See section 3.3.2.

In regard to claim 4, OMSOFT further discloses:

4. The environment of claim 1 further comprising: a plurality of management objects,

each said management object being associated with at least one of said objects; a human

manager object; and an interface for network evolution for coupling said management

objects to said human manager object, wherein said human manager object manages

said objects through said management objects. See page 109, section 6.4 paragraph 1.

In regard to claim 5, OMSOFT further discloses:

5. The environment of claim 4 wherein said human manager object assigns increasing

object version numbers to said objects. See page 16 item 3.

In regard to claim 6, OMSOFT further discloses:

6. The environment of claim 5 wherein said human manager object assigns

monotonically increasing interface versions to said object interfaces wherein each said

object interface has a unique global identification in said application network. See page

91, section 5.9.1.

In regard to claim 7, OMSOFT further discloses:

7. The environment of claim 6 further comprising: means for determining said

compatible interfaces of said objects by registering said global identification and said

object version number of said object with said management object. See section 5.9.1.

In regard to claim 8, OMSOFT further discloses:

8. The environment of claim 7 further comprising: means for determining an object table

comprising rows representing said object versions of said objects in said network

application and columns representing an object identification and interface

identification; means for sorting said determined object table with respect to an object

version; means for sorting a first said sorted object table for a first said object and a

second said sorted object table for a second said object with respect to a common said

interface identification; means for joining said first said sorted object table and said

second said sorted object with respect to said interface identification; and means for

extracting said compatible object from said join of said object tables. See example on

pages 93-98.

In regard to claim 9, OMSOFT further discloses:

9. The environment of claim 8 further comprising: means for sorting a subsequent object

table with respect to said common said interface identification; and means for joining

said subsequent object table with said joined first said sorted object table and said

second said sorted object table. See example on pages 93-98.

In regard to claim 10, OMSOFT further discloses:

10. The environment of claim 1 further comprising a life cycle framework including a

specification stage in which said objects and said interfaces are specified, a design stage

in which said interfaces of said objects are negotiated, an implementation stage in which

said negotiated interfaces of said objects are implemented and a testing stage in which

said implemented interfaces are tested. See section 3.4 on pages 57-59.

In regard to claim 11, OMSOFT discloses:

11. A method for implementing negotiation during software development comprising the

steps of:

a. determining a human manager object; See page 15, section 1.4.3:

The human manager has a manager object...

> b. *determining at least one management object;* See pages 15 and 16, section 1.4.3:

>> The manager object and the management object share an interface (Interface for Network Evolution – INE) through which the human manager sends appropriate commands to the management object, to configure and control the application network and its evolution.

> c. *determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object* (see pages 15 and 16, section 1.4.3 as cited directly above), *by instructing objects with said at least one management object to create a plurality of objects for performing object operations, each said object including an object interface,* See page 21 "CreateObject/DestoryObject":

>> Human manager sends this command...

> d. *creating an interaction means for connecting said at least one object to said management objects;* See page 42, section 3.2.3:

>> It is a group interaction mechanism where interaction proceeds...

> e. *determining at least one management object port associated with said management object; f. determining at least one object port associated with said object;* See section 3.2, 3.2.1, and 3.2.2 as cited in the above rejection of claim 1.

> g. *forwarding negotiations <scripts> from said object ports to said management object ports.* See section 8.2: "Tetherless Negotiation Between Developers", especially the bottom of page 183:

>> D1 writes this information to the port and then releases and triggers the negotiation port.

In regard to claims 12-18, OMSOFT further discloses:

12. The method of claim 11 further comprising the step of : assigning tasks of designing said objects from said human manager object to a respective developers associated with at least one of said objects.

13. The method of claim 12 further comprising the step of : creating a developer negotiation port by said developer for each of said objects to be developed.

14. The method of 13 further comprising the step of : registering said developer negotiation ports with said human manager object.

15. The method of claim 14 further comprising: creating management negotiation ports at said management objects which are each associated respectively with one of said developer negotiation ports.

16. The method of claim 15 wherein step g comprises: forwarding negotiation scripts written in modified CORBA IDL by said developers through said respective developer negotiation ports to said respective manager negotiation ports for forwarding to designated objects.

17. The method of claim 16 further comprising the step of : forwarding said scripts written in modified CORBA IDL received at said management object to said human manager object via said INE.

18. The method of claim 17 further comprising the step of : interpreting said script written in modified CORBA IDL received at said human manager object into human readable data.

See section 8.2 on pages 180-186.

In regard to claim 19, OMSOFT further discloses:

19. The method of claim 11 wherein the step of forwarding negotiations is repeated until all developers have agreed. See page 184, paragraph 1.

In regard to claim 20, OMSOFT further discloses:

20. The method of claim 19 wherein said negotiations determine an object interface defined in modified CORBA IDL. See section 3.3.2 on page 55.

In regard to claim 21, OMSOFT discloses:

21. A method for implementing a network application comprising the steps of:

determining a plurality of objects; associating an object port with each of said objects; See sections 3.2, 3.2.1, and 3.2.2 as cited above in the rejection of claim 1.

determining transactions for exchanging messages between said objects; See page 14 paragraph 1:

> Developers write an IDL description that describes the structure, the protocol
> (independent transactions)...

determining an object interface for each said object; and implementing each determined object interface, wherein said messages are exchanged sequentially between said objects having compatible said object interfaces. See sections 3.2, 3.2.1, and 3.2.2 as cited above in the rejection of claim 1.

In regard to claim 22, OMSOFT further discloses:

22. The method of claim 21 further comprising the step of : registering said implemented

object and said object interface with a management framework, said management

framework returning an object identification and an object version identification and an

interface version identification. See section 5.13.1 on pages 100 and 101.

In regard to claim 23, OMSOFT further discloses:

23. The method of claim 22 wherein said implementing step further comprises the step of

: determining a network application having compatible said object version

identifications. See section 5.9.1

In regard to claim 24, OMSOFT further discloses:

24. The method of claim 23 wherein said step of determining a network application

having compatible object versions comprises the steps of : a. determining an object table

comprising rows representing said object identification and said object version

identification and columns representing said interface version identification; b. sorting

said determined object table with respect to said object version identification; c. sorting a

first said sorted object table for a first said object and a second said sorted object table

for a second said object with respect to a common said interface identification; d. joining

said first said sorted object table and said second said sorted object with respect to said

interface identification to form a join of said object tables; and e. extracting said

compatible object from said join of said object tables. See example on pages 93-98.

In regard to claim 25, OMSOFT further discloses:

25. The method of claim 24 further comprising the steps of : f. sorting a subsequent object table with respect to said common said interface identification; and g. joining said subsequent object table with said joined object table of step (d); See example on pages 93-98.

In regard to claim 26, OMSOFT further discloses:

26. The method of claim 24 wherein said object tables are created to have said object version identification and said interface version identification increasing in said rows and said columns. See page 91, section 5.9.1 and Figure 5-3.

In regard to claim 27, OMSOFT further discloses:

27. The method of claim 21 further comprising the steps of : determining a plurality of management objects, each said management object being associated with at least one of said objects; determining a human manager object; determining an interface for network evolution for coupling said management objects to said human manager object; and managing said objects by said human manager object through interacting with said management objects. All limitations have been addressed in the above rejection of claim 11.

In regard to claim 28, OMSOFT further discloses:

28. The method of claim 27 further comprising the steps of : updating said determined

objects; and assigning increasing object version numbers by said human manager object

to said updated objects through said management objects. See section 5.13.1 on pages

100 and 101.

In regard to claim 29, OMSOFT further discloses:

29. The method of claim 27 further comprising the step of : updating said object

interface; and assigning increasing interface version numbers by said human manager

object to said updated object through said management objects. See section 5.13.1 on

pages 100 and 101.

In regard to claim 30, OMSOFT discloses:

30. A method for setting up a network application comprising the steps of:

a. determining a human manager object; b. determining at least one management

object; c. determining an interface for network evolution (INE) between said human

manager object and said management object, by said human manager object; instructing

said at least one management object by said human manager object to create at least one

object for performing object operations, each said object including an object interface, d.

creating an interaction means for connecting said objects to said management objects,

said interaction means also being connected to said INE and said human manager

object; These limitations have been addressed in the above rejection of claim 11.

e. initializing states at said human manager object of said objects and forwarding said initialized states to said objects via said INE to forward to said initialized states to said management object and said management object forwarding said initialized states from said management object to said objects. See page 119, section 6.5: "Initialization of object states"

In regard to claim 31, OMSOFT further discloses:

31. The method of claim 30 after step c further comprising the steps of : f determining a human manager object INE port for said human manager object; g. determining a management object INE port for said management object; and h. associating said INE with said INE port for said management object and said INE port for said manager object. See Figure 6-1.

In regard to claim 32, OMSOFT further discloses:

32. The method of claim 31 further comprising the steps of : determining at least one port associated with said management object; and determining at least one object port associated with each said object. See Figure 6.1.

In regard to claim 33, OMSOFT further discloses:

33. The method of claim 30 wherein said object interface is defined in modified CORBA IDL. See section 3.3.2 on page 55

In regard to claim 34, OMSOFT discloses:

34. A method for dynamically reconfiguring a network application comprising the steps of:

determining a human manager object; determining at least one management object; determining an interface for network evolution (INE) between said human manager object and said management object, by said human manager object, by instructing objects with said at least one management object to create at least one object for performing object operations, each said object including an object interface and having an original state, creating an interaction means for connecting said at least one object to said management objects; determining at least one management object port associated with said management object; These limitations have been addressed in the above rejection of claims 11 and 30.

establishing quiesent points in at least one of said objects to be reconfigured through said management object. See page 123, Section 6.6.7:

> Here the management waits until the application gives on its own (voluntarily) same points needed for dynamic reconfiguration.

In regard to claim 35, OMSOFT further discloses:

35. The method of claim 34 further comprising the step of : forwarding data for updating said at least one object from said object to said human manager object. See section 6.6.7 on page 123.

In regard to claim 36, OMSOFT further discloses:

36. The method of claim 35 further comprising the steps of : determining said port of said object to be reconfigured; sending a destroy command from said human manager object to destroy said port to be reconfigured; creating a new version of said object to be reconfigured at said human manager object; forwarding said new version of said object to said management object; creating a new object having said new version of said object; and determining a new object port associated with said new object. See section 6.6.10 on pages 126-129.

In regard to claim 37, OMSOFT further discloses:

37. The method of claim 36 further comprising the steps of : determining at said human manager object if a said original state of said object is the same as a state of said new version of said object; and if said original object version and said new version have the same states, replacing said original object version with said new version; or if said original object version and said new version do not have the same state, determining at said human manager object an equivalent state and replacing said original version with said new version. See page 127 item 1.

In regard to claim 38, OMSOFT further discloses:

38. The method of claim 37 further comprising the step of forwarding data for updating said at least one interface version from one of said objects to said human manager object. See page 126 paragraph 1.

In regard to claim 39, OMSOFT further discloses:

39. The method of claim 38 further comprising the steps of : determining a number of said objects to be reconfigured for said updating of said interface version; sending a destroy command from said human manager to destroy said number of objects to be reconfigured; creating a new version of each said number of objects to be reconfigured at said human manager object; forwarding said new versions to said management object; and creating a corresponding number of new objects having said new versions. See pages 126-129.

In regard to claim 40, OMSOFT further discloses:

40. The method of claim 34 wherein said object interface is defined in modified CORBA IDL. See section 3.3.2 on page 55.

In regard to claims 41-45, all limitations have been addressed in the above rejections of claims 11, and 13-16, respectively.

In regard to claims 46-50, all limitations have been addressed in the above rejections of claims 21-24 and 20, respectively.

In regard to claims 51-54, all limitations have been addressed in the above rejections of claims 31-33, respectively.

In regard to claims 55-58, all limitations have been addressed in the above

rejections of claims 34, 36, 39, and 40, respectively.

Conclusion

Any inquiry concerning this communication or earlier communications from the

examiner should be directed to J. Derek Rutten whose telephone number is (571) 272-3703. The

examiner can normally be reached on T-F 6:00 - 4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the

organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

jdr

TUAN DAM
SUPERVISORY PATENT EXAMINER